

PATENT APPLICATION

**USER INTERFACE FOR A DIGITAL PRODUCTION SYSTEM
INCLUDING MULTIPLE WINDOW VIEWING AND NAVIGATING**

Inventor(s): Sukendeep Samra, a citizen of United Kingdom, residing at
2500 Abbott Kinney Blvd., Apt. 1
Venice, CA 90292

Makoto Kawamura, a citizen of Japan, residing at
1-3-8-107 Shirokane
Minato-ku
Tokyo 108-0072 Japan

Shingo Kawabe, a citizen of Japan, residing at
13603 Marina Pointe Dr., #A504
Marina del Rey, CA 90292

Kyoko Nakamura, a citizen of Japan, residing at
13701 Marina Pointe Drive, #306
Marina del Rey, CA 90292

Assignee: Sony Corporation and Sony Electronics, Inc.

Entity: Large

USER INTERFACE FOR A DIGITAL PRODUCTION SYSTEM INCLUDING MULTIPLE WINDOW VIEWING AND NAVIGATING

CLAIM OF PRIORITY

[01] This application claims priority from the following co-pending U.S. Provisional Patent Applications:

[02] 1. Application Ser. No. 60/271,376; filed 2/22/01, entitled "A System and Method for Editing" (client docket 50P4410); and

[03] 2. Application Ser. No. 60/284,660; filed 4/17/01, entitled "Advanced System and Method for Editing" (client docket 50R4639).

CROSS-REFERENCE TO RELATED APPLICATIONS

[04] This application is related to the following co-pending U.S. Patent Applications:

[05] 1. Application Ser. No. 09/665,721; filed 9/18/00, entitled "System And Method For Optimizing The Processing Of Images" (client docket 50N3647);

[06] 2. Application Ser. No. 09/691,795; filed 10/18/00, "System And Method For Increasing Performance When Compositing Images" (client docket 50N3649);

[07] 3. [Ser. No. and filing date TBA] entitled "User Interface for Generating Parameter Values in Media Presentations Based on Selected Presentation Instances" (attorney docket 20699-45);

[08] 4. [Ser. No. and filing date TBA] entitled "Collaborative Computer-Based Production System Including Annotation, Versioning and Remote Interaction" (attorney docket 20699-46); and,

[09] 5. [Ser. No. and filing date TBA] entitled "Media Production System Using Flowgraph Representation of Operations" (attorney docket 20699-47);

BACKGROUND OF THE INVENTION

[10] This invention relates in general to digital processing systems and more specifically to a digital processing system using a graphical representation of operations to advantageously accomplish processing of digital media productions.

[11] Today's digital processing systems are used in many applications. One intensive application area uses digital production systems to create and process movies, video, animation, audio and other types of digital media. Such systems are referred to as production systems.

[12] The demand on a production system is extremely high. A huge amount of information is present in even a few frames (less than one second) of a visual production, such as a movie. The type and number of possible complex operations that can be performed on a movie require enormous processing power. Production systems provide incredible flexibility in applying different operations such as effects, transitions, editing, adjustments and other modifications. Each of dozens, or more, types of operations may in themselves have many different parameters, or other characteristics, which can be modified. The selection and application of operations on hundreds of selected areas of frames, clips and other parts of a production has become a highly specialized and complex task.

[13] For example, a typical operation may be to composite two images together. The decision to perform a composite operation leads a human user of a production system down a path of myriad choices such as selecting frames or clips for compositing, cropping and scaling the images, performing color correction, erosion, blurring, light balancing, creating masks, etc. Several sets of operations may be required for different portions within a single frame, or screen area of a clip.

[14] In present systems, the user interface (i.e., the display and user input devices) of productions systems is of great importance. Any improvement in ease of use and efficiency is often realized as a huge benefit in decreasing production times and providing better production results. However, the design of such user interfaces is difficult because the production system is usually limited to a relatively small display area (e.g., 1 or two display screens) to both display and play back the media, and to provide controls for navigating among thousands of production parts and for applying the complex operations.

[15] Thus, it is desirable to provide an invention that improves upon the prior art user interfaces in production systems.

BRIEF SUMMARY OF THE INVENTION

[16] The present invention provides viewing features for an applications program such as a digital image processing program, non-linear editor, post-production system, etc. One aspect of the invention provides different ways for users to select an object and an action to be applied to the object. The user can select an object first and then an

action, or vice versa. The user can also use gestural input to designate both an object and action virtually simultaneously.

[17] Another aspect of the invention provides multiple views of a same image. Each view, or window, can be independently sized, zoomed, panned, etc. Any effects performed on the image appear in all of the windows since each window shows (potentially) different portions of the same image content.

[18] Another aspect of the invention provides a navigation window to help a user move within a large image or diagram that does not fit entirely on a single display screen. The navigation window includes an inner box that shows, in miniature, the principal objects in the screen display. A region outside of the screen display is shown in the navigation box, relative to the inner box.

[19] Another aspect of the invention derives an active area from multiple operations mark in/out points. Operations that are part of a group have their in/out points adjusted so that the operation does not occur outside of the group's own mark in/out points.

[20] In one embodiment the invention provides a method for performing actions on objects, the method executing in a computer system including a processor coupled to a user input device, the method comprising

[21] accepting signals from a user input device to first specify a first object and then select a first action to be performed on the first object; accepting signals from a user input device to first select a second action and then specify a second object on which the second action is to be performed; and accepting signals from a user input device to define a gesture that overlaps a third object, wherein the gesture is mapped to a third action to be performed on the third object.

[22] In another embodiment the invention provides a method for displaying images on a display screen, the method comprising displaying multiple windows on the display screen; performing an operation on an image;

[23] displaying the image in each of the multiple windows; and accepting input from a user input device to allow independent manipulation of the windows.

[24] In another embodiment the invention provides a method for viewing an image on a display screen, wherein the display screen is coupled to a processor and user input device, the method comprising

[25] displaying a navigator box on the display screen; displaying a miniature version of the image on the display screen within an inner box within the navigator box on the display screen, wherein the inner box is smaller than the navigator box, wherein

portions of the image not displayed on the display screen are shown in miniature within the area of the navigator box that is outside of the inner box.

[26] In another embodiment the invention provides a method for determining active intervals of operations to be performed on images, wherein each operation includes a start time and a stop time that defines an initial active interval for the operation, the method comprising selecting one or more operations to be members of a group; determining a start time and a stop time to define a group interval for the group; and setting the active region of each operation that is a member of the group to be the intersection of each operation's initial active interval with the group interval.

[27] The present invention provides features whereby image content in a media production system is associated with a visual flowgraph representation of the operations used to create, or derive, the image content. A comprehensive association is maintained between production parts in image space and in flowgraph space. This allows a user to quickly move between working with an image representation of a part and the flowgraph (i.e., visual display of operations) representation.

BRIEF DESCRIPTION OF THE DRAWINGS

[28] Fig. 1A shows a first screen display of the user interface of the present invention;

[29] Fig. 1B illustrates the correspondence of flowgraph representations to image representations;

[30] Fig. 1C illustrates the maintenance of correspondence between parts and flowgraphs;

[31] Fig. 1D is an illustration of a computer system suitable for use with the present invention;

[32] Fig. 1E illustrates subsystems that might typically be found in a computer such as the computer system of Fig. 1D;

[33] Fig. 2A shows a flowchart illustrating basic steps in a routine to implement object/action processing;

[34] Fig. 2B shows flowchart to illustrate basic steps of a routine to perform action/object processing;

[35] Fig. 2C shows a flowchart illustrating basic steps of a routine to perform gestural processing;

- [36] Fig. 3A shows a main window in a working panel of an image processing system;
- [37] Fig. 3B illustrates the use of multiple windows;
- [38] Fig. 3C shows the effect of resizing four windows;
- [39] Fig. 3D illustrates the effect of scaling, rotating and panning operations in the windows; Fig. 4A illustrates navigator viewing as applied to viewing a relatively large flowgraph;
- [40] Fig. 4B shows navigator and multiple window viewing;
- [41] Fig. 5A shows a flowgraph including a group node;
- [42] Fig. 5B shows the result of expanding a group node; and
- [43] Fig. 5C illustrates start and stop times of operations in a group.

DETAILED DESCRIPTION OF THE INVENTION

[44] A specific, preferred embodiment, production system is referred to as "Socratto" which is developed and marketed by VFX, a division of Sony Corporation. Although aspects of the present invention are described in connection with the Socratto system, it should be apparent that the invention is suitable for use with many different productions systems having different ways to present information and to accept user inputs. Also, although the invention is described herein primarily with reference to film or video production systems, the invention is applicable to other types of media production systems such as computer-aided design, audio, animation, modeling, etc. Various aspects of the invention are applicable to any type of application program or functional system.

[45] Fig. 1A shows a first screen display of the user interface.

[46] In Fig. 1A, display screen 110 is used to display first panel area 116 and second panel area 114. In the preferred embodiment, first panel area is used to display images, clips and a "flowgraph" diagram depending on the current activity of the user. Second panel area 114 includes user interface controls that can change, as needed, in accordance with the display and operation of the first panel area. The preferred embodiment of the invention uses the lower panel as the primary control area so that the more substantial portion of the display screen defined by the first panel is available for frame and clip playback and for display of flowgraphs (which can become complex). Naturally, any arrangement of panels, orientation of display areas and controls, or other use of one or more display screens to effect a user interface display is possible.

[47] Fig. 1B illustrates the correspondence of flowgraph representations to image representations.

[48] In Fig. 1B, image space 120 is a conceptual category that includes visual presentation of images. Such images can be bitmap, vector representation, photographic projection or other type of digital or non-digital data formats. Image space is useful to present a large amount of visual information to a human user, as where a frame is displayed or a clip is played back. Because of the innate ability of humans to process very large amounts of visual information almost instantaneously, image space representation is an efficient and important way for users to, e.g., compare image results, predict anticipated changes and necessary operations, determine when a desired result is achieved, etc. Further, image space representation is the ultimate form that a completed production assumes.

[49] Frames, or images, such as frame 132, 136 and 140 exist in image space. Clips typically include multiple frames and also exist in image space. Clips 102, 104 and 110 are shown in Fig. 1B. Clip section 108 is also shown, along with other frames and clip portions, not referenced. In a preferred embodiment, groups of clips and frames can be organized into a "reel." The frames and clips in Fig. 1B can be considered part of a single reel, but any type of organization is possible. In general, although reference is made to specific types of production parts such as frames, images, clips, clip sections, frame layers, etc., any production part, group of parts, or portion of a part can be included.

[50] Flowgraph space 130 is a second conceptual category that includes a visual presentation of operations that are performed to generate a production part. In a preferred embodiment, flowgraphs are associated with clips or clip sections (including single frames or portions of frames). The associations can be one-to-one, one-to-many or many-to-one; although a preferred embodiment maintains a one-to-one relationship between each unique clip section and the flowgraph that generated it. Any manner of correspondence or associations among production parts and clips are possible.

[51] In Fig. 1B, flowgraph 112 is associated with clip 102. Clip 104 does not have a flowgraph associated with it. Frame 106 is associated with flowgraph 116. Clip section 108 is associated with flowgraph 118. Clip 110 is not associated with flowgraph 118 since clip 110 is merely used as an input to create clip 108, as discussed, below.

[52] Each flowgraph's associated part represents the output of the operations that the flowgraph describes. For example, In Fig. 1B, flowgraph 112 starts with clip 134 and performs operations on the clip as represented by the oval nodes and connectors of the flowgraph. Flowgraph diagrams are discussed in more detail, below. The output of

flowgraph 112 is represented in flowgraph space as node 135 of flowgraph 112. The output is represented in image space as clip 102. The input clip 134 can be any type of part, as can the resulting output. In other words, a single frame can be used to generate multiple frames, a frame portion or layer can be used to generate a full frame, clip, etc.

[53] Flowgraph 116 illustrates the case where the output of a flowgraph, namely flowgraph 116, is a single frame, frame 106

[54] Flowgraph 118 illustrates the case where two media sources (e.g., from computer RAM, disk storage, a digital network, optical scanning device, etc.) are used to product clip section 108. Clip 110 represents one of the media sources. The other media source is not shown. The dashed arrow indicates that clip 110 is not associated with flowgraph 118 since clip 110 is not generated by flowgraph 118.

[55] The correspondence, or association, of data items between image space and flowgraph space can be by any means as is known in the art. Parts can be stored or represented in any image format. The flowgraphs can be stored or represented as any type of suitable data structure such as a list, relational database elements, etc. The association between images and flowgraphs can be made with pointers, lists, embedding a part with an associated flowgraph in a file, or by any suitable programming or data processing technique.

[56] Fig. 1C illustrates the maintenance of correspondence between parts and flowgraphs during any and all types of operations performed in the production system of a preferred embodiment. In a preferred embodiment, there is always a flowgraph counterpart to every production part. Other embodiments need not maintain such a strict rule. However, the insurance of a flowgraph for each part means that a user can easily and instantly switch between viewing and manipulating media in the image space and editing the media in flowgraph space. This achieves a unification of visual presentation of the media with visual presentation of operations used to create the media. As is discussed below, this approach (in concert with other features) provides many benefits.

[57] Fig. 1C shows user interfaces used to perform manipulations to parts in image space. For example, image interface 160 can be used to change the color of an image. The user can then switch to an interface in flowgraph space, represented by flowgraph interface 162, to visually work with the operations. The color change operation is automatically inserted into the flowgraph so that the flowgraph-to-image correspondence is maintained. In a preferred embodiment, every operation that changes a part results in corresponding modifications to the flowgraph associated with the part so that each part

always has an up-to-date flowgraph that visually shows the operations used to create or derive the image.

[58] In the course of a typical production, many operations can be made to a part both in image space and in flowgraph space. This is illustrated in Fig. 1C by additional image interface 164 and additional flowgraph interface 166. In the preferred embodiment the user is provided with controls to easily switch back and forth between interfaces in the two spaces.

[59] All types of operations are paralleled. Storage/retrieval and transfer operations are symbolized at 168 and 170. When a part or flowgraph is stored, the associated flowgraph or part, respectively, is also stored in association so another session, program, process, etc., can retrieve the part-flowgraph pair. Note that the two types of representation need not be stored in the same logical or physical area or device. Also, portions of each representation can reside at different location, on different devices, etc., as desired. Similar to storage, when a part or flowgraph is transferred (e.g., over a network to a remote system), the counterpart representation is also made available to the destination.

[60] Retrieval and transfer of a representation pair is illustrated by interfaces 172 and 156.

[61] Thus, the maintenance of closely-linked and continuously updated image and flowgraph pair representations provides advantages in a production system.

[62] Fig. 1F is an example of a flowgraph according to the present invention.

[63] In Fig. 1F, nodes such as 180, 182, 184, 186 and 188 represent operations. Operations can be effects, filters, processes, actions or any type of action performed on, or with, media content being produced. A list of representative operations is shown in Table I. It should be apparent that many additional types of operations are suitable for use with the present invention.

- | | |
|--------------------------|--|
| 1. Brightness Adjustment | control of luminance in RGB and Alpha channels. |
| 2. Clamp Effect | restricts pixel values within determined ranges. |
| 3. Contrast | adjusts the range between brightest and darkest tones. |
| 4. Convert | change pixel types between, e.g., 8 bit, 16 bit floating and working. Can also change image types between RGB, RGBA and Alpha channel outputs. |
| 5. Crop | allows trimming an image on the left, right, top or bottom. |
| 6. Dissolve | produces an image that is a mixture of input images. |
| 7. Fade | produces an image that is a mixture of an input image and a solid color. |

8. Gain	adjusts the brightness of individual channels of a image or clip.
9. Gamma	adjusts the apparent brightness by altering the middle range of brightness.
10. Invert	reverses an image's color and shades so that black becomes white, white becomes black, red becomes cyan, etc.
11. CMY Graph	adjusts the Cyan, Magenta and Yellow channels.
12. Luma Graph	adjusts the luminance of an image.
13. Monochrome	converts a color image into a monochrome image by adjusting color saturation in each of the color channels.
14. Offset	adjusts the brightness of the individual channels of an image.
15. Swap RGBA	shifts any color channel to become any other color channel.

TABLE I

[64] Nodes have names that provide brief descriptions, or identifications, of the type of operation represented by the node. For example, node 180 has a name, "Color," indicating that the node is a source for a predefined color output. Output port 190 is an output port for node 180. Connector 194 connects the output of node 180 to node 182 by connecting to node 182's input port 192. Node 182 is called "Clip_Output" which indicates the output of the flowgraph operations. In other words, the flowgraph output is generated at "Clip_Output."

[65] The present invention uses a "user input model" that provides an efficient way for a user to specify an action to be performed on an object. Objects in the system such as frames and clips in image space, or nodes representing effects or other operations in flowgraph space, can have an action applied. For example, an action, such as a color correction operation, can be applied to an object, such as a frame.

[66] The system allows the user to specify object-action processing in three different ways. The first is "object/action" where an object is first identified and then an action is selected to apply to the object. A second type of object-action processing is "action/object" where an action is first selected and then an object to which to apply the action is identified. Finally, "gestural" object-action processing allows choosing an object and action at essentially the same time.

[67] Fig. 2A shows flowchart 200 illustrating basic steps in a routine to implement object/action processing.

[68] After the routine is entered, step 202 is executed to accept user input from an input device to specify an object. The object can be specified by any manner as is

known in the art. For example, the object can be "clicked on" with a mouse and pointer, pen tablet, touch screen, trackball and pointer, etc. A pointer can select an object semi-automatically as where a pointer is moved into the vicinity of an object. See, for example, co-pending patent [Ser. No. and filing date TBA] entitled "Media Production System Using Flowgraph Representation of Operations" (attorney docket 20699-47) referenced, above. The object can be scrolled to the center of the display screen to a "focus" area to become a specified object. The object can be a default object such as some previously specified object that is no longer on-screen. The object can be specified by name, as, for example, where the user types in a name or other identifier for an object. Other ways of specifying an object are possible.

[69] In a preferred embodiment, once a user has specified an object, the object remains highlighted until an action is applied, another object is specified, or the specified object becomes de-specified. Multiple objects can be specified at once, or successively. Thus, the approach of selecting an object before an action allows multiple objects to easily be applied with one or more actions.

[70] After an object is specified at step 202 of Fig. 2A, additional signals are accepted from a user input device to select an action to apply to the object. Actions can be selected in many ways. For example, actions can be selected in the same way that objects are specified, above, if the action is visually indicated on the display screen. Ways to visually indicate an action on the display screen include showing node images, icons, lists of action names, etc.

[71] After an action is selected, step 206 is executed to choose a destination. The destination can be by default -- in which case there is no active step to choose the destination. For example, the destination can be the same location as the object. In other words, if the object is a clip stored on a disk, the destination can be to the same disk location to overwrite the original clip. Usually, the destination is preferably a different location in the same storage device as the original object, although any destination is possible. The user can specify a device destination as the default. For example, remote storage such as a redundant array of inexpensive disks (RAID) can be used over a network such as a local-area network (LAN) or the Internet.

[72] After the object, action and destination have been obtained, the routine of flowchart 200 exits so that the system can perform the object-action process as directed.

[73] Fig. 2B shows flowchart 210 to illustrate basic steps of a routine to perform action/object processing.

[74] In Fig. 2B, Step 212 accepts signals from a user input device to select an action. Next step 214 allows user specification of an object (or objects). This differs from the flowchart in Fig. 2A in that an action is selected before an object. This allows several actions to be pre-selected for application to a single object. Also, the pre-selected actions can be applied repeatedly to different objects. Similar to the flowchart of Fig. 2A, a destination is chosen at step 216.

[75] Fig. 2C shows flowchart 220 to illustrate basic steps of a routine to perform gestural processing.

[76] In Fig. 2C, step 222 is executed to allow simultaneous, or single-action, selection of both an object and an action. One approach is to allow a user to "draw" on top of a displayed object. A single character, or symbol, is interpreted as a "gesture" to determine the type of action to apply. For example, drawing a "C" on the image of a frame object on the display indicates that a color correction operation is to be performed on the frame object. Many other types of distinguishable symbols can be used to indicate other actions. Similarly, where an object ID is drawn onto a visual indication of an operation (e.g., onto a "node" displayed on the display device) then the object becomes the subject of the action written upon. Step 224 is executed to choose a destination, if necessary, similar to the above.

[77] Figs. 3A-D show the quad viewing mode of the present invention.

[78] In Fig. 3A, screen display 240 shows main image window 242 occupying upper panel area 244. Although a user will often work with a single main window as shown in Fig. 3A, a "quad" viewing and drawing mode is provided where four instances of an image are shown in four different adjacent windows.

[79] Fig. 3B illustrates four windows 246, 248, 250 and 252 that fill the upper panel area. The four instances of the same image are useful to view different areas of the image simultaneously to determine whether a desired effect has been achieved with any applied operations. Each window can be panned and zoomed independently. Windows can also be resized independently.

[80] Fig. 3C shows the effect of resizing the four windows of Fig. 3B.

[81] In Fig. 3C, window 254 has been shortened by moving its right boundary to the left. This results in an expansion of window 256 to the left. Window 256 has also been expanded downwards at the expense of windows 258 and 260. Note that the movement of the window boundaries results in a clipping of the image in the bounded window.

[82] In Fig. 3D, different scaling, rotating and panning operations have been performed from Fig. 3C in order to show different results in each window. The image in window 262 has been rotated. The image in window 264 has been scaled up (i.e., zoomed in). The image in window 266 has been scaled down. The image in window 268 remains the same as the image in window 260 of Fig. 3C.

[83] Throughout all of the independent window and image manipulations, each image in the quad view is processed with any operations and effects in the same way. In other words, the bitmap representations of the content of each window is the same except for the viewing operations -- which do not affect the image content. Thus, the user is provided with a convenient way to view the image simultaneously at different scales. The user can view effects in a very small area and also view the larger picture to see the image to the full scale, as it would appear in the final production. Naturally, any number of windows can be provided.

[84] A navigator drawing mode provides a small "wide-area" indicator to help a user determine what portion of a larger image or diagram is being viewed on the display screen.

[85] Fig. 4A illustrates navigator viewing as applied to viewing a relatively large flowgraph.

[86] In Fig. 4A, display screen 300 shows flowgraph portion 301 including nodes 302, 304, 306, 308 and 310; with various connectors to connect nodes shown in the display screen and nodes off-screen (not shown). As is apparent from Fig. 4A, not all of the flowgraph is shown in display screen 300.

[87] Navigator box 312 includes miniature representation 314 of display screen 300. Miniature representation, or "mini screen," 314 includes the major items that can be seen on display screen 300. Namely, the nodes of visible flowgraph portion 301 are shown on the mini screen as 316, 318, 320, 322 and 324, respectively. Note that node 326 which is shown in navigator box 312 but not in mini screen 314 is an item that is part of the larger flowgraph diagram but which is not visible on display screen 300. Thus, the user can tell from the navigator box that scrolling the display screen view upwards will bring another node, corresponding to node 326, within view.

[88] Navigator box 312 is preferably large enough to include viewing space around all four sides of mini screen 314 so that the user can effectively see what will come into view if the display screen view is scrolled in different directions.

[89] Another feature of the present invention combines the navigator view with the quad view as shown in Fig. 4B.

[90] In Fig. 4B, windows 340, 342, 344 and 346 provide the quad view as described above. Navigator box 360 now shows four min screens to correspond with each of the windows. Unlike the discussion above for quad views, the four windows of Fig. 4B are now used to view four different parts of a diagram so their image content is not the same.

[91] The four different parts of a larger diagram are shown relative to each other in the diagram as mini windows 350, 352, 354 and 356 corresponding with windows 340, 342, 344 and 346, respectively. As each window is scrolled, zoomed, etc., the corresponding mini window is changed accordingly and the navigator box is manipulated to accommodate the display of all four mini windows.

[92] The present invention uses the intersection of effects within a common group to determine the effective start/stop points to apply an effect.

[93] Fig. 5A shows a flowgraph where group node 422, called "Group1," is connected between nodes 420 and 424.

[94] Fig. 5B shows the result of expanding Group1 into its group nodes, or sub-nodes. Group nodes "Gamma" and "Blur" are now connected between nodes 420 and 424. Each node can be assigned a start and stop time to designate the bounds in which the node's effect, or operation, will be applied to a clip.

[95] Fig. 5C further illustrates start and stop times of group nodes.

[96] In Fig. 5C, clip 400 has three different operations, or effects, applied to the clip at different times. Each effect has a start and stop point, also referred to as "mark in" and "mark out" points. Effects are also referred to, and represented by, nodes.

[97] In Fig. 5C, Gamma node 402 is an effect that starts at frame A and ends at frame B. Blur node 404 is an effect that starts at frame C and ends at frame D. A feature of the present invention allows any number of nodes to be "grouped" into another common node. When grouped, all nodes in the group are connected and disconnected in accordance with the single group node's connections on a flowgraph that includes the group node. Group nodes can also be present within other group nodes. Each group node can have its own start and stop points. The clip section between a node's start and stop points is the active area for the node's operation.

[98] A preferred embodiment of the invention uses the intersection of overlapping active areas to determine the active area for a sub-nodes of a group. In the example of Fig. 5C, Group1's active area is E-F. However, Gamma node 402's active area is

E-B. This is only a portion of Gamma node 402's active area, A-B, were it not part of Group1. In other words, each sub-node's active area is the intersection, or overlap, of the sub-node's active area with the group node's active area. Blur node 404's active area remains C-D which is the full extent of Blur node 404's active area even as a standalone (i.e., not part of a group) node since all of Blur node 404's standalone active area is within the active area of Group1.

[99] To further illustrate. If Group1 node is later made part of Group2, then all of the active areas are now intersected with Group2's active area, G-H, in order to obtain the nodes' active areas as part of Group2. Namely, Gamma node 402's active area in Group2 is G-B and Blur node 404's active area in Group2 is G-D. Group1's active area while in Group2 is G-F.

Basic Hardware

[100] Figs. 1D and 1E illustrate basic hardware components suitable for practicing the present invention.

[101] Fig. 1D is an illustration of computer system 1 including display 3 having display screen 5. Cabinet 7 houses standard computer components (not shown) such as a disk drive, CDRom drive, display adapter, network card, random access memory (RAM), central processing unit (CPU), and other components, subsystems and devices. User input devices such as mouse 11 having buttons 13, and keyboard 9 are shown. Other user input devices such as a trackball, touch-screen, digitizing tablet, etc. can be used. Other media presentation devices other than display screen 5 may be used such as projection displays, wearable computer monitor (e.g., "dataglass"), etc. In general, the computer system is illustrative of but one type of computer system, such as a desktop computer, suitable for use with the present invention. Computers can be configured with many different hardware components and can be made in many dimensions and styles (e.g., laptop, palmtop, pentop, server, workstation, mainframe). Any hardware platform suitable for performing the processing described herein is suitable for use with the present invention.

[102] Fig. 1E illustrates subsystems that might typically be found in a computer such as computer 1.

[103] In Fig. 1E, subsystems within box 20 are directly interfaced to internal bus 22. Such subsystems typically are contained within the computer system such as within cabinet 7 of Fig. 4A. Subsystems include input/output (I/O) controller 24, System Memory (or random access memory "RAM") 26, central processing unit CPU 28, Display Adapter 30, Serial Port 40, Fixed Disk 42, Network Interface Adapter 44. The use of bus 22 allows each

of the subsystems to transfer data among subsystems and, most importantly, with the CPU. External devices can communicate with the CPU or other subsystems via bus 22 by interfacing with a subsystem on the bus. Thus, Monitor 46 connects with Display Adapter 30, a relative pointing device (e.g. a mouse) connects through Serial Port 40. Some devices such as Keyboard 50 can communicate with the CPU by direct means without using the main data bus as, for example, via an interrupt controller and associated registers.

[104] As with the external physical configuration shown in Fig. 1D, many subsystem configurations are possible. Fig. 1E is illustrative of but one suitable configuration. Subsystems, components or devices other than those shown in Fig. 1E can be added. A suitable computer system can be achieved without using all of the subsystems shown in Fig. 1E. For example, a standalone computer need not be coupled to a network so Network Interface 44 would not be required. Other subsystems such as a CDROM drive, graphics accelerator, etc. can be included in the configuration without affecting the performance of the system of the present invention.

[105] Any suitable programming language can be used to implement the routines of the present invention including C, C++, Java, assembly language, etc. Different programming techniques can be employed such as procedural or object oriented. The routines can execute on a single processing device or multiple processors. Although the flowchart format demands that the steps be presented in a specific order, this order may be changed. Multiple steps can be performed at the same time. The flowchart sequence can be interrupted. The routines can operate in an operating system environment or as stand-alone routines occupying all, or a substantial part, of the system processing.

[106] Steps can be performed by hardware or software, as desired. Note that steps can be added to, taken from or modified from the steps in the flowcharts presented in this specification without deviating from the scope of the invention. In general, the flowcharts are only used to indicate one possible sequence of basic operations to achieve a functional aspect of the present invention.

[107] Although the present invention is described predominantly with respect to a visual image processing system, many types of systems that include media presentation features can employ the present invention. Such systems include, but are not limited to, audio, animation, modeling, computer-aided design, web browsing or access, etc. Other embodiments need not devote screen area to specific controls, such as the controls at 114 of Fig. 1A. It should be apparent that the entire screen, or different portions of the screen, can be used for media presentation and parameter modification as described herein.

[108] Although the invention has been discussed in connection with parameters and parameter values, other structures for modifying media instances, such as attribute/value pairs, arrays, lists, etc., can be employed. In general, a parameter, or other data structure, used to describe an image need not only have a single value. Multiple values, data structures, functions, etc., can be used to define parameter values either continuously or discretely. For example, where a function is used to define parameter values over space, time, etc., the present invention can check for different functions associated with shared parameter types as opposed to merely checking for different values, as described above.

[109] Thus, the scope of the invention is to be determined solely by the appended claims.